

## Data base

A Data base is simply an organized Place where data (information) is stored so you can access and use it easily without confusion.

\* A Data base is a collection of related data.

- Storing → تخزين أسماء طلاب ودرجاتهم بدل الورق
- Retrieve → تبيث عن درجة طالب واحد وسط آلاف الطلاب
- UPdate → تستبدل طالب جديد في الجدول
- Add → لو طالب غير عنوانه نعمل تعديل
- Delete → لو طالب اخرج نصح بياناته

|         | id   | name | Phone       | Grade       | salary      | Dfoc        |
|---------|------|------|-------------|-------------|-------------|-------------|
| Student | 1    |      |             |             | null values | null values |
|         | 4000 |      |             |             |             |             |
| Staff   | 4001 |      |             | null values |             | null values |
|         | 4500 |      |             |             |             |             |
| Depts   | 4501 |      | null values | null values | null values |             |
|         | 4510 |      |             |             |             |             |

بتخزن مساهمة كل الفاقف



STU-STAFF

نظري entities  $\Rightarrow$  attributes "Filed"

جس table

Column

P.K "P.K"

①

"P.K"

③

①

| Students | Did | Sid  | name | Ph  |   | id  | name | Salary | Did |
|----------|-----|------|------|-----|---|-----|------|--------|-----|
|          | --- | 1500 | ---  | --- | m | 1   | ---  | ---    | 2   |
|          | --- | 1503 | ---  | --- |   | --- | ---  | ---    | 1   |
|          | --- | 2500 | ---  | --- |   | --- | ---  | ---    | 2   |
|          | --- | 2501 | ---  | --- |   | --- | ---  | ---    | 3   |
|          | --- | 3000 | ---  | --- |   | --- | ---  | ---    | 1   |
|          | --- | 3001 | ---  | --- | 1 | --- | ---  | ---    | 2   |
|          | --- | 4000 | ---  | --- | m | 500 | ---  | ---    | 5   |

| Depts | Did | name | loc |
|-------|-----|------|-----|
|       | 1   | IT   | --- |
|       | 2   | CS   | --- |
|       | 3   | IS   | --- |
|       | 4   | DS   | --- |

Relations

- one to one
- one to many
- many to many

• 1 To m

m To 1  $\rightarrow$  m To m

• 1 To 1

1 To 1  $\rightarrow$  1 To 1

• 1 To m

m To 1  $\rightarrow$  1 To m



# Data Type

- أرقام → int
- لعدد → number
- حروف → Char
- عنوان → address → var Char To
- تاريخ → date

• Column الذي يميز كل Column ⇒ unique/Not null ⇒ Primary Key

التي تأتيه من جدول ال one ونقطه P.K  
في جدول ال many

⇒ P.K ⇒ ممكن / عادي يتكرر / هيربط الجدول الأول بالجدول الثاني  
التي ينتقل بنفس ال "data Type" يكون null

⇒ P.K ⇒ unique/Not null  
يكون null

⇒ 1 To 1 ⇒ بنفس ال Columns على بعضها

⇒ m To m ⇒ لازم نكسوها X  
في الداتا بيز ⇒ هنعمل مكانها جدول جديد

التي يكون عبارة عن ال P.K للجدولين

⇒ ممكن مثلا نثبت الطالب والدكاترة نتيجي

وممكن العكس ⇒ فيه تكرار عادي



## Data Base users

- Database  $\Rightarrow$  A Collection of related data
- Data  $\Rightarrow$  Known facts that can be recorded and have an implicit meaning
- Mini-world (universe of discourse (UOD)) :- <sup>جامعة موجودة</sup> <sup>فعلية</sup> Some Part of real world about which data is stored in a database  $\Rightarrow$  <sup>في درجات طلاب في الجامعة</sup>
- Data base Management system (DBMS)  $\Rightarrow$  <sup>بومبي</sup> <sup>891</sup> A software system to facilitate the creation and maintenance of a computerized database.
- Database system  $\Rightarrow$  the DBMS software together with the data itself. Sometimes, the applications and are also included



- Defining a Database  $\Rightarrow$  specifying data types, structures and constraints for the stored data.

بيانات وصفية

- Metadata  $\Rightarrow$  information that describes the database itself (like table and file names), stored by the DBMS in a Catalog or Dictionary.

- Manipulating a database  $\Rightarrow$

- Retrieval  $\Rightarrow$  Querying and generating reports.
- Modification  $\Rightarrow$  insertions, deletions, and updates.
- Accessing database through web applications

- Sharing a Database  $\Rightarrow$  Multiple users access it at once
- Application Program  $\Rightarrow$  Sends queries to the DBMS
- Query  $\Rightarrow$  Retrieve data

الوصول على بيانات  
له استعلام طلب

## \*\* Application interact with databases through:

- ✓ Queries  $\Rightarrow$  Retrieve and display data.
- ✓ Transactions  $\Rightarrow$  Read, update, add data

## \*\* Application must $\Rightarrow$

- Prevent unauthorized access. تمنع الوصول غير المصرح به
- Adapt to change user needs.



## Protection includes

« SYSTEM Protection »

« Security Protection »

↳ Against hardware or software failures

↳ Against unauthorized or malicious access

### \*\* Maintain the database system :-

- let it evolve with changing requirements.
- Applications must adapt to user needs

### Main characteristics of the Database Approach ⇒

1) self describing:

↳ DBMS stores both data and metadata.

2) Program - data independence:

↳ Data structure changes don't affect programs

3) Data abstraction:

↳ Hides storage details, show conceptual view. إظهار مفهومي

4) Multiple views:

↳ Each user sees only relevant data. تدوين مستخدم يرى البيانات

5) Data sharing & multi-user processing: الوقت

↳ Many users can access data safely at once,

Concurrency and recovery ensure correctness.

لأنه يسمح لكثير من المستخدمين في نفس الوقت



# Database users

## 1) ACTORS on the Scene:

المشاركون في الواقعة

- DBA ⇒ Manage access, Performance, and Resources  
مدير القاعدة
- Designers ⇒ Define structure and constraints.
- End users ⇒ use the database.
  - ↳ Casual → occasional use عاديون
  - ↳ Naive → use Pre-defined functions بسيط ← موففونك
  - ↳ Sophisticated → experts using advanced tools متقدمون
  - ↳ Standalone → use Personal database مستقلون
- System Analysts & Developers ⇒ design and build APPs  
حلالة النظام
- Business Analysts ⇒ analyze data for decisions.  
محللو الأعمال

## 2) workers behind the scene:

الكاملون خلف الكواليس

- System Designers/Implementors: build DBMS software  
مهندس النظام
- Tool Developers: Create tools for DB design and monitoring  
مطور الأدوات
- Operating Staff: run and maintain system.  
المشغلون وموففوا البيانات



Advantage of using DS  $\Rightarrow$ 

- 1) less data redundancy  $\rightarrow$  data stored once, Consistent  
نقل البيانات
- 2) Better security  $\rightarrow$  Controlled access by DBA  
أمان الوصول
- 3) Persistent storage  $\rightarrow$  data and objects saved Permanently  
تخزين دائم
- 4) Efficient queries  $\rightarrow$  indexing and optimization.  
الاستعلام فعال
- 5) Backup & recovery  $\rightarrow$  data Protection  
النسخة الاحتياطية
- 6) Multiple interfaces  $\rightarrow$  easy user access  
واجهات متعددة
- 7) Handles complex relationships  
تقريب العلاقات المعقدة
- 8) Maintains data integrity.  
الحفاظ على سلامة البيانات
- 9) Supports rules & triggers for automation  
دعم القواعد والتبقيات التلقائية

Additional implications  $\Rightarrow$ 

- 1) Standard enforcement  $\Rightarrow$  Consistent data format.  
توحيد المعايير
- 2) Faster development  $\Rightarrow$  Quicker app creation  
تطوير أسرع
- 3) Flexibility  $\Rightarrow$  easy structure change  
مرنة
- 4) Availability of current information  $\Rightarrow$  real time data access  
معلومات حديثة
- 5) Economic of Scale  $\Rightarrow$  less resource waste  
توفير في الموارد



# Data Models, Schemas, and Instances

## 1. Data abstraction

يعطى نظرة عن البيانات بشكل مجرد ونحذف التفاصيل التي مشاهدين يبتغونها

- Suppression of details of data organization and storage
- Highlighting of the essential features for an improved understanding of data

«مختب»  
تفصيل

التنظيم

والمتخزين»

# Data Models, Schemas, and Instances (cont'd.)

## 2. Data model

data type  
Relation

- Collection of concepts that describe the structure of a database
- Provides means to achieve data abstraction
- Basic operations
  - Specify retrievals and updates on the database.

### • the dynamic aspect or behavior of a database application.

- This allows the database designer to specify a set of valid user-defined operations that are allowed on the database objects. An example of a user-defined operation could be COMPUTE\_GPA, which can be applied to a STUDENT object.

مجموعة من المفاهيم  
التي أفترض وجودها  
بها الدان



# Categories of Data Models

## 1. High-level or conceptual data models →

- Close to the way many users perceive data

بشكلها  
بالصور التي

## 2. Low-level or physical data models →

بفهمها  
الصور

- Describe the details of how data is stored on computer storage media

فيها لك التفاصيل

## 3. Representational data models implementation

- Easily understood by end users
- Also similar to how data organized in computer storage

سهل  
الصور التي  
فهمها  
تفاصيل  
بشكلها

# Categories of Data Models (cont'd.)

## (1) High-level or Conceptual data models

### • Entity

- Represents a real-world object or concept

### • Attribute

- Represents some property of interest
- Further describes an entity

### • Relationship among two or more entities

- Represents an association among the entities
- Entity-Relationship model

كائن / دكتور

مجموعة  
بيانات حقيقية

name  
Phone  
address

علاقة / رابط  
بين الكائنات  
Entities



# Categories of Data Models (cont'd.)

→ Representational / implementational

- Relational data model

- Used most frequently in traditional commercial DBMSs
- Represent data by using record structures and hence are sometimes called record-based data models.

Record.



# Categories of Data Models (cont'd.)

## (2) Low-level or Physical data models

- Describe how data is stored as files in the computer by representing information such as record formats, record orderings, and access paths
- Access path**
  - Structure that makes the search for particular database records efficient
- Index**
  - Example of an access path
  - Allows direct access to data using an index term or a keyword

بيته  
بالخاويل

الطريق الذي يسهل عشان  
البيانات

قوله  
مباشرة  
للحالة التي انت عاوزها  
زي خفرك لدا

## Schemas, Instances, and Database State

- Database schema**
  - Description of a database which is specified during database design.
  - is not expected to change frequently

### Schema diagram

- Displayed schema

The diagram displays the structure of each record type but not the actual instances of records.

### Schema construct

- Each object in the schema such as STUDENT or COURSE

### Database state or snapshot

- Data in database at a particular moment in time

database

Person 1  
Person 2  
موايل  
شلا

ايه الـ database  
Constraints  
وهكذا

تغير نادر

مش بيوضح البيانات  
ولكن بيوضح البناء  
بناع الجدول

Description  
↓  
schema

وهي حالات بتغير كل لحظة



# Database Schema vs. Database State

- Database State: →

- Refers to the content of a database at a moment in time. حالة البيانات في وقت معين
- The actual data stored in a database at a particular moment in time. This includes the collection of all the data in the database.
- Also called database instance (or occurrence or snapshot).

- The term instance is also applied to individual database components, e.g. record instance, table instance. سجل معين

- Initial Database State:

- Refers to the database state when it is initially loaded into the system. الحالة الأولى
- Every time an update operation is applied to the database, we get another database state. كل الحالة يتغير

- Valid State:

- A state that satisfies the structure and constraints of the database.

مثلا  
age 70

valid

لوفظ

invalid



# Schemas, Instances, and Database State (cont'd.)

- Schema evolution → لما المتطلبات تتغير لازم نعدل في قاعدة البيانات نفسها
  - Changes applied to schema as application requirements change

## ■ Distinction

- The database schema changes very infrequently. → بتغير بشكل نادر
- The database state changes every time the database is updated  
↓  
بتغير مع كل مرة نعدل أو نضيف بيانات
- Schema is also called intension. →
- State is also called extension



# Three-Schema Architecture and Data Independence

- The goal of the three-schema architecture is to separate the use applications from the physical database.

لـ يفصل اللي بيظهره الليوزر  
عن الجداول

## 1. Internal level بيهم بالتفصيل

- Describes physical storage structure of the database. «low level»
- The internal schema uses a physical data model and describes the complete details of data storage and access paths for the database.

## 2. Conceptual level

«أخفب ليهم  
المستخدم»

Representational data  
«model»

- has a conceptual schema
- Describes structure of the whole database for a community of users
- The conceptual schema hides the details of physical storage structures and concentrates on describing entities, data types, relationships, user operations, and constraints.

## 3. External or view level «بيهم الليوزر»

- Describes part of the database that a particular user group is interested in and hides the rest of the database from that user group.

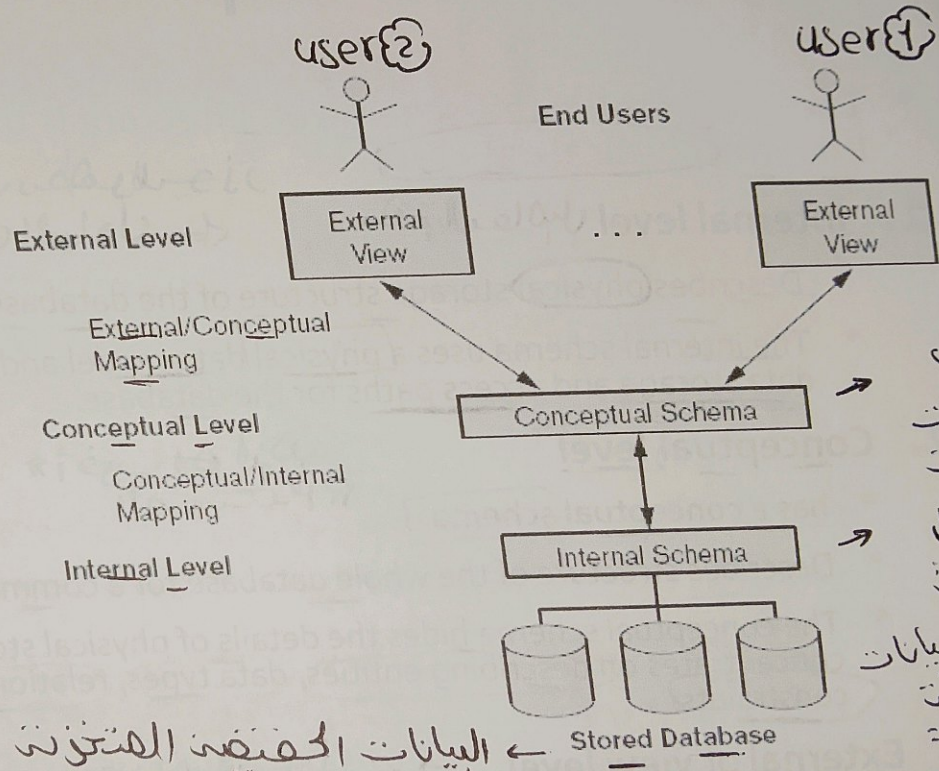
لـ الحالة اللي الليوزر مهتم به فقط

بيتم مثلا موظف الأمن ميوصله علاقة بـ الطالب Schema



# Three-Schema Architecture and Data Independence (cont'd.)

**Figure 2.2**  
The three-schema architecture.



جداول  
علاقات  
كيانات  
تفاصيل  
تقنية  
إدارة البيانات  
مستخرجة  
فعل  
على الهارد

البيانات الحقيقية المستخرجة  
فعل في ال Hard  
أو ال Memory

DBMS  
↓  
يتولى الطلب  
External View  
ال Internal  
ويعيد يدير البيانات  
ويغير شكلها  
عشان تطابق  
External View



# Three-Schema Architecture

« العلاقات التي  
توجد بين  
المستويات »

- Mappings among schema levels are needed to transform requests and data.
- Programs refer to an external schema, and are mapped by the DBMS to the internal schema for execution.
- Data extracted from the internal DBMS level is reformatted to match the user's external view (e.g. formatting the results of an SQL query for display in a Web page)

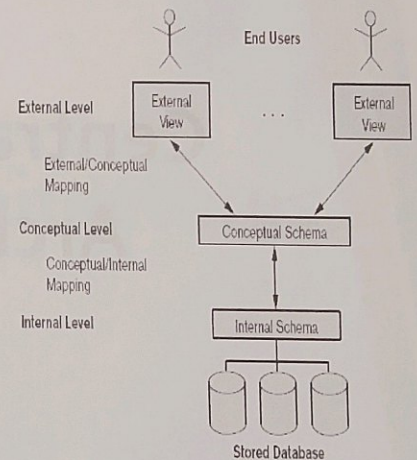
إضافة  
جدول  
في ذلك  
طريقة  
البنية

Logical ⇒ قدرتي تغيير الـ conceptual  
external الـ تغيير الـ  
Physical ⇒ قدرتي تغيير في الـ internal  
EX / Con الـ تغيير في الـ

## Data Independence

إمكانية  
تغيير الـ schema في level one

Figure 2.2  
The three-schema architecture.



- Capacity to change the schema at one level of a database system

level  
بغير  
واحدة

- Without having to change the schema at the next higher level
- Types:

1. **Logical data independence** is the capacity to change the conceptual schema without having to change external schemas or application programs
2. **Physical data independence** is the capacity to change the internal schema without having to change the conceptual schema. Hence, the external schemas need not be changed as well.



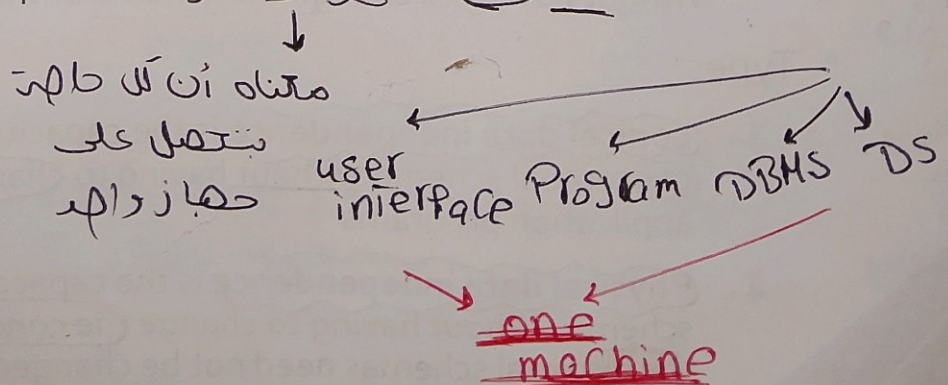
# DBMS Languages

- Data definition language (DDL)  
→ بنیاد الداتا بین • Defines both schemas
- Storage definition language (SDL)
  - Specifies the internal schema
- View definition language (VDL)
  - Specifies user views/mappings to conceptual schema
- Data manipulation language (DML)
  - Allows retrieval, insertion, deletion, modification

## Centralized and Client/Server Architectures for DBMSs

### • Centralized DBMSs Architecture

- All DBMS functionality, application program execution, and user interface processing carried out on one machine





# Basic Client/Server Architectures

العميل client Server الخادم

الجهاز  
التي يقدم  
الخدمة

الجهاز  
التي يطلب  
الخدمة

## • Servers with specific functionalities

### • File server

- Maintains the files of the client machines.

مسؤول  
عن تخزين  
البيانات

### • Printer server

- Connected to various printers; all print requests by the clients are forwarded to this machine

مسؤول بطابعات

السيرفر يستقبل

### • Web servers or e-mail servers

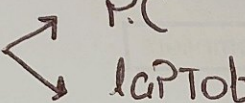
الطلب ويرسله للطباعة  
المناسبة

يستقبل طلبات المواقع  
من السيرفرز  
ويحولها لمواقع  
الويب

إرسال  
واستقبال الرسائل  
بين ال users



# Basic Client/Server Architectures (cont'd.)

- Client machines  P.C  
laptop

- Provide user with:

- Appropriate interfaces to utilize these servers
- Local processing power to run local applications

بـ وحدة معالجة

كافية ومباشرة

كل شوية

هنتحتاج الـ بورد

مستوفز  
والله



# Basic Client/Server Architectures (cont'd.)

## ✓ Client

- User machine that provides user interface capabilities and local processing

## • Server

- System containing both hardware and software
- Provides services to the client machines
  - Such as file access, printing, archiving, or database access

Client → machine → local / شخصي  
Processing  
STEP دون استخدام ال Server

## Two-Tier Client/Server Architectures for DBMSs

« استخدام لغة SQL »

- Server handles ↗
  - Query and transaction functionality related to SQL processing
- Client handles
  - User interface programs and application programs





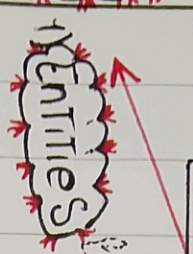


# Three-Tier and n-Tier Architectures for Web Applications

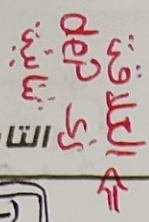
- **Application server or Web server**
  - Adds intermediate layer between client and the database server
    - Runs application programs and stores business rules
- N-tier
  - Divide the layers between the user and the stored data further into finer components



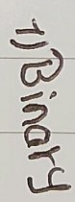
lec 4



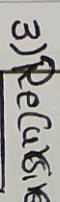
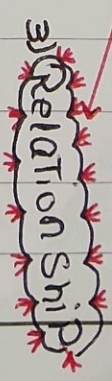
الحلقة - ٥٥٨٩



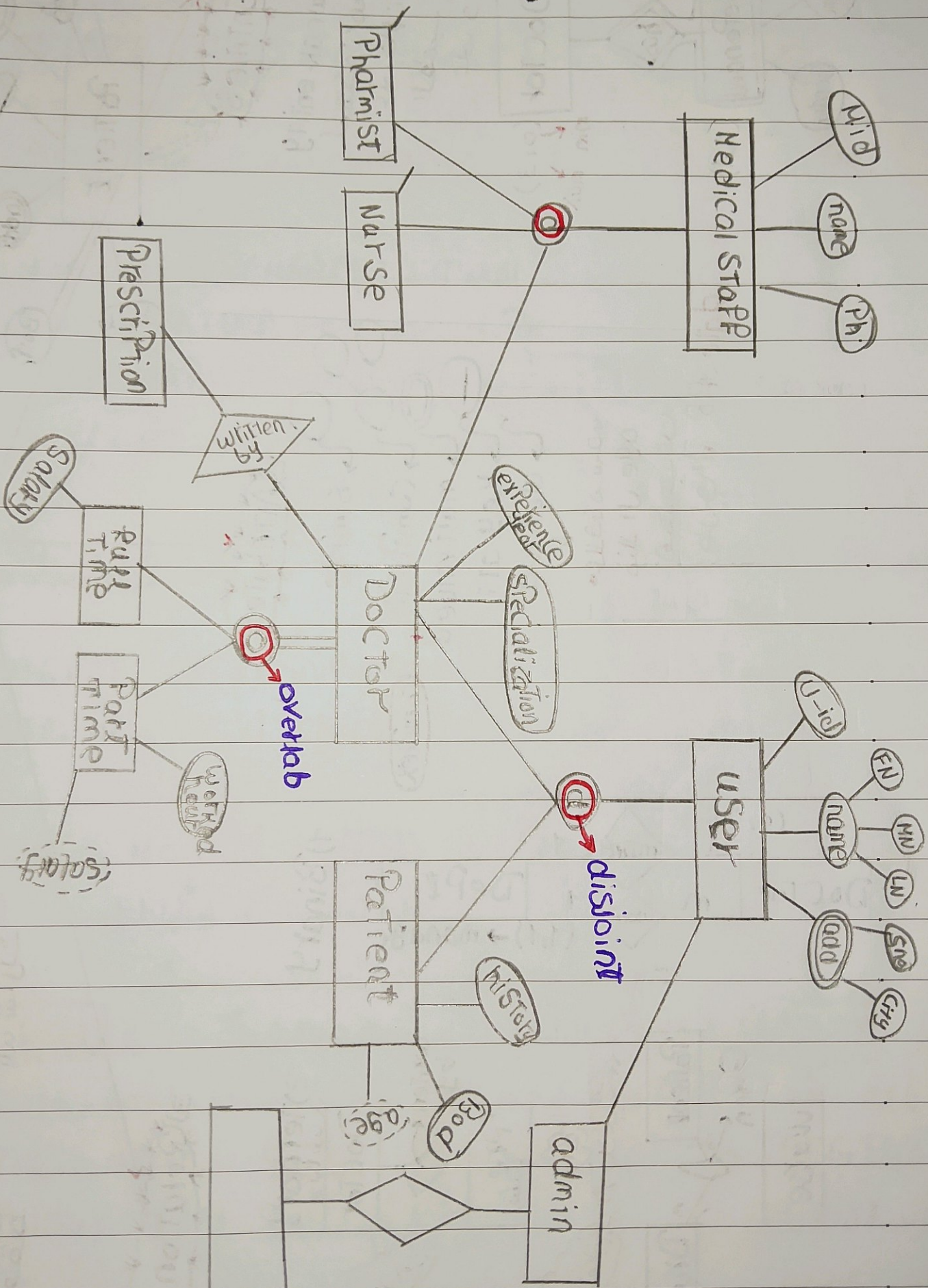
Complex



5. 5. 5.









class variable → Programming و Database علاقة بين ال

⇒ class ⇒ Doctor , Table ⇒ Doctor

⇒ variable ⇒ name , Attribute ⇒ name

↓  
« Programming »

↓  
« Database »

inheritance

⇒ Super class ⇒ الأب ويكون فيه كل الصفات المشتركة

⇒ Sub class ⇒ الابن والـ هيرث + هيكون له صفاتها لو فيه

⇒ users

Super ⇒ name / id / address « حاجات مشتركة »

Sub ⇒ الاختلافات } حاجات مختلفة  
⇒ Doctor ⇒ specialist  
⇒ Patient ⇒ age

Specialization ⇒ Super و هتقسموا ل Subs  
Generalization ⇒ Subs في Super

• disjointness ⇒ « هل ينفع ال Class بيدلوا الأدوار »

لـ مثلا هل الدكتور ينفع يكون المريفين ؟؟

لـ دائما لو انت دكتور مينفعش في نفس الوقت

تكون للمريفين

• overlapping ⇒ « عادي مثلا ال Part-time يشتغل J Full-time »

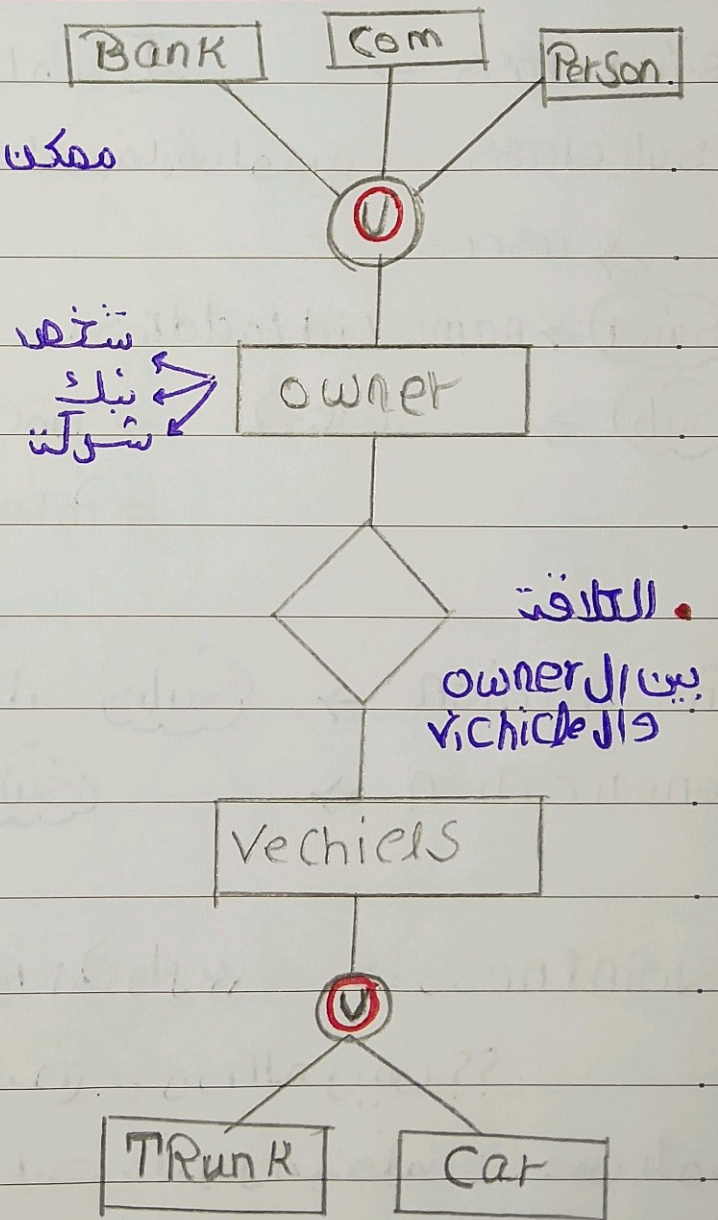
والعكس دائما حسب القاعدة الـ عندك



- Completeness
    - Partial → Patient / Dr مثلا فيه Admin غير ال
    - Total → Full time / Part time مفيش حاجه غير
- «هنا فيه حاجه ثابتة غير الكتب مثلا»

- Sub Class لوله سوپر كلاسه واحد
- Lattice ⇒ Sub Class لوله أكثر من سوپر كلاسه

Union ⇒ ممكن تجمع حاجات مختلفه في المئات





# Recursive Relation

foreign key

الطبيعي انك بتعمل constrain من جدول ال many  
بيشاور على جدول ال one

Recursive ← هيشاور على نفسه

Create table Doctor (

Did integer Primary Key,

Name Varchar2 (200),

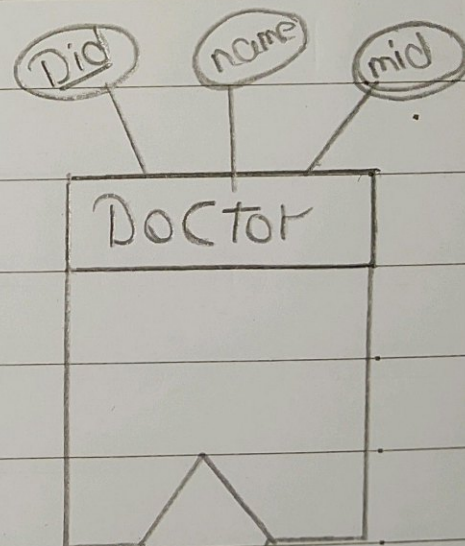
Mid integer,

constrain FK Mid foreignkey (Mid) reference Doctor

(Did)

Manager

);



«Super Vission»